

日 本 国 特 許 庁
JAPAN PATENT OFFICE

31. 5. 2004

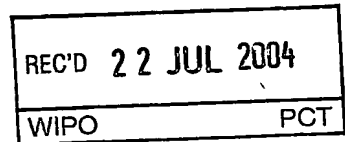
別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日
Date of Application: 2 0 0 3 年 5 月 2 8 日

出 願 番 号
Application Number: 特 願 2 0 0 3 - 1 5 0 0 8 8
[ST. 10/C]: [J P 2 0 0 3 - 1 5 0 0 8 8]

出 願 人
Applicant(s): 三 洋 電 機 株 式 会 社

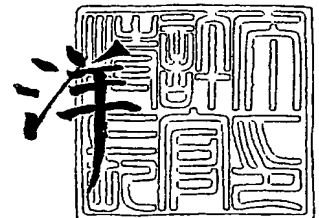


PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

2 0 0 4 年 7 月 8 日

特許庁長官
Commissioner,
Japan Patent Office

小 川



【書類名】 特許願
【整理番号】 NQB1030047
【提出日】 平成15年 5月28日
【あて先】 特許庁長官殿
【国際特許分類】 G02B 27/00
H04N 13/00

【発明者】

【住所又は居所】 大阪府守口市京阪本通2丁目5番5号 三洋電機株式会社内

【氏名】 竹本 賢史

【発明者】

【住所又は居所】 大阪府守口市京阪本通2丁目5番5号 三洋電機株式会社内

【氏名】 濱岸 五郎

【発明者】

【住所又は居所】 大阪府守口市京阪本通2丁目5番5号 三洋電機株式会社内

【氏名】 増谷 健

【発明者】

【住所又は居所】 大阪府守口市京阪本通2丁目5番5号 三洋電機株式会社内

【氏名】 吉川 隆敏

【発明者】

【住所又は居所】 大阪府守口市京阪本通2丁目5番5号 三洋電機株式会社内

【氏名】 堀 吉宏

【発明者】

【住所又は居所】 大阪府守口市京阪本通2丁目5番5号 三洋電機株式会社内

【氏名】 堀内 啓次

【特許出願人】

【識別番号】 000001889

【氏名又は名称】 三洋電機株式会社

【代表者】 桑野 幸徳

【代理人】

【識別番号】 100105843

【弁理士】

【氏名又は名称】 神保 泰三

【手数料の表示】

【予納台帳番号】 067519

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0011478

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 立体映像表示装置及びプログラム

【特許請求の範囲】

【請求項 1】 ファイルに基づいて立体視映像を生成する立体映像表示装置であって、ファイル内の記述のなかで立体視用処理を示す記述部分を判断する手段と、立体視用処理を示す記述部分に基づいて対象オブジェクトの位相ずらし量及びずらし方向を判断する手段と、前記位相ずらし量及びずらし方向に基づいて対象オブジェクトの各視点画像の描画処理を行う手段と、を備えたことを特徴とする立体映像表示装置。

【請求項 2】 請求項 1 に記載の立体映像表示装置において、対象オブジェクトの位相ずらし量及びずらし方向に応じて、前記対象オブジェクトをその隣接のオブジェクト上に上書き描画するか又は前記対象オブジェクト上にその隣接のオブジェクトを上書き描画することを特徴とする立体映像表示装置。

【請求項 3】 請求項 2 に記載の立体映像表示装置において、上書き描画される対象オブジェクトを半透明化処理することを特徴とする立体映像表示装置。

【請求項 4】 請求項 2 又は請求項 3 に記載の立体映像表示装置において、前記上書き描画処理はそれを示す記述がファイル内に存在するときに実行することを特徴とする立体映像表示装置。

【請求項 5】 請求項 1 に記載の立体映像表示装置において、対象オブジェクトの各視点画像について、対象オブジェクトの隣側のオブジェクトの位置を、対象オブジェクトのずらし方向の側に、位相ずらし量以上の分だけ、ずらして描画することを特徴とする立体映像表示装置。

【請求項 6】 請求項 5 に記載の立体映像表示装置において、前記隣側のオブジェクトの位置のずらし描画処理はそれを示す記述がファイル内に存在するときに実行することを特徴とする立体映像表示装置。

【請求項 7】 請求項 1 乃至請求項 6 のいずれかに記載の立体映像表示装置において、ファイルに位相ずらし量を示す情報として程度情報が記述されている場合に、予め保持している設定テーブルの情報と前記程度情報とから位相ずらし量を算出することを特徴とする立体映像表示装置。

【請求項 8】 請求項 1 乃至請求項 7 のいずれかに記載の立体映像表示装置において、対象オブジェクトが手前側に立体視されるときには対象オブジェクトを拡大描画処理し、奥側に立体視されるときには対象オブジェクトを縮小描画処理することを特徴とする立体映像表示装置。

【請求項 9】 コンピュータを、ファイル内の記述のなかで立体視用処理を示す記述部分を判断する手段と、立体視用処理を示す記述部分に基づいて対象オブジェクトの位相ずらし量及びずらし方向を判断する手段と、前記位相ずらし量及びずらし方向に基づいて対象オブジェクトの各視点画像の描画処理を行う手段として機能させるためのプログラム。

【請求項 10】 請求項 1 に記載のプログラムにおいて、コンピュータを、対象オブジェクトの位相ずらし量及びずらし方向に応じて、前記対象オブジェクトをその隣接のオブジェクト上に上書き描画するか又は前記対象オブジェクト上にその隣接のオブジェクトを上書き描画する手段として機能させるためのプログラム。

【請求項 11】 請求項 10 に記載のプログラムにおいて、コンピュータを、上書き描画される対象オブジェクトを半透明化処理する手段として機能させるためのプログラム。

【請求項 12】 請求項 10 又は請求項 11 に記載のプログラムにおいて、コンピュータを、上書き描画を行うことを示す記述がファイル内に存在するときに前記上書き描画処理を実行する手段として機能させるためのプログラム。

【請求項 13】 請求項 9 に記載のプログラムにおいて、コンピュータを、対象オブジェクトの各視点画像について、対象オブジェクトの隣側のオブジェクトの位置を、対象オブジェクトのずらし方向の側に、位相ずらし量以上の分だけ、ずらして描画する手段として機能させるためのプログラム。

【請求項 14】 請求項 13 に記載のプログラムにおいて、コンピュータを、隣側のオブジェクトの位置のずらし描画処理を行うことを示す記述がファイル内に存在するときに隣側のオブジェクトの位置のずらし描画処理を実行する手段として機能させるためのプログラム。

【請求項 15】 請求項 9 乃至請求項 14 のいずれかに記載のプログラムにお

いて、コンピュータを、ファイルに位相ずらし量を示す情報として程度情報が記述されている場合に、予め保持している設定テーブルの情報と前記程度情報とから位相ずらし量を算出する手段として機能させるためのプログラム。

【請求項16】 請求項9乃至請求項15のいずれかに記載のプログラムにおいて、コンピュータを、対象オブジェクトが手前側に立体視されるときには対象オブジェクトを拡大描画処理し、奥側に立体視されるときには対象オブジェクトを縮小描画処理する手段として機能させるためのプログラム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】

この発明は、立体映像表示装置及びプログラムに関する。

【0002】

【従来の技術】

立体視技術としては、従来よりパララックスバリアを用いる眼鏡無し立体視方法、偏光眼鏡や液晶シャッタ眼鏡などを用いる眼鏡有り立体視方法などが知られている。また、立体視させる映像についても、実写の映像だけでなく、コンピュータグラフィックスを用い、仮想空間上に配置したオブジェクトを平面に投影して描画処理する3D描画による映像がある。更には、前記描画処理を二視点において行なうことで、右眼映像と左眼映像を作成することができる。また、2次元映像信号から抽出された奥行き情報と2次元映像信号とに基づいて立体映像を生成する立体映像受信装置及び立体映像システムが提案されている（特許文献1参照）。2次元映像と奥行き情報とからなる映像ファイルを作成すれば、このファイルを開いたときに、立体映像を生成することができる。また、二つの映像を1チャンネルの映像として放送し、受信機側で立体視が行なえる方法が提案されている（特許文献2参照）。二つの映像からなる映像ファイルを作成すれば、このファイルを開いたときに、立体映像を生成することができる。

【0003】

【特許文献1】

特開2000-78611号公報

【特許文献2】

特開平10-174064号公報

【0004】

【発明が解決しようとする課題】

ところで、ファイルにはHTMLファイルなどのように、文字のみ或いは文字と画像（ビットマップ画像，gif画像，JPEG画像など）からなるファイルが存在し、このようなファイルにおいて任意の文字や画像部分について立体視させることが望まれる。

【0005】

この発明は、上記の事情に鑑み、HTMLファイルなどのファイルに基づいて任意の文字部分や画像部分を立体表示させることができる立体映像表示装置及びプログラムを提供することを目的とする。

【0006】

【課題を解決するための手段】

この発明の立体映像表示装置は、上記の課題を解決するために、ファイルに基づいて立体視映像を生成する立体映像表示装置であって、ファイル内の記述のなかで立体視用処理を示す記述部分を判断する手段と、立体視用処理を示す記述部分に基づいて対象オブジェクトの位相ずらし量及びずらし方向を判断する手段と、前記位相ずらし量及びずらし方向に基づいて対象オブジェクトの各視点画像の描画処理を行う手段と、を備えたことを特徴とする（以下、この項において第1構成という）。

【0007】

上記第1構成の立体映像表示装置において、対象オブジェクトの位相ずらし量及びずらし方向に応じて、前記対象オブジェクトをその隣接のオブジェクト上に上書き描画するか又は前記対象オブジェクト上にその隣接のオブジェクトを上書き描画するようにしてもよい。また、上書き描画される対象オブジェクトを半透明化処理するようにしてもよい。前記上書き描画処理は上書き描画を行うことを示す記述がファイル内に存在するときに実行するようにしてもよい。

【0008】

また、第1構成の立体映像表示装置において、対象オブジェクトの各視点画像について、対象オブジェクトの隣側のオブジェクトの位置を、対象オブジェクトのずらし方向の側に、位相ずらし量以上の分だけ、ずらして描画するようになっていてもよい。また、前記隣側のオブジェクトの位置のずらし描画処理は、位置ずらし描画を行うことを示す記述がファイル内に存在するときに実行するようになっていてもよい。

【0009】

これらの構成の立体映像表示装置において、ファイルに位相ずらし量を示す情報として程度情報が記述されている場合に、予め保持している設定テーブルの情報と前記程度情報とから位相ずらし量を算出するようにしてもよい。また、対象オブジェクトが手前側に立体視されるときには対象オブジェクトを拡大描画処理し、奥側に立体視されるときには対象オブジェクトを縮小描画処理するようになっていてもよい。

【0010】

また、この発明のプログラムは、コンピュータを、ファイル内の記述のなかで立体視用処理を示す記述部分を判断する手段と、立体視用処理を示す記述部分に基づいて対象オブジェクトの位相ずらし量及びずらし方向を判断する手段と、前記位相ずらし量及びずらし方向に基づいて対象オブジェクトの各視点画像の描画処理を行う手段として機能させることを特徴とする（以下、この項において第2構成という）。

【0011】

上記第2構成のプログラムにおいて、コンピュータを、対象オブジェクトの位相ずらし量及びずらし方向に応じて、前記対象オブジェクトをその隣接のオブジェクト上に上書き描画するか又は前記対象オブジェクト上にその隣接のオブジェクトを上書き描画する手段として機能させるようにしてもよい。また、コンピュータを、上書き描画される対象オブジェクトを半透明化処理する手段として機能させるようにしてもよい。また、コンピュータを、上書き描画を行うことを示す記述がファイル内に存在するときに前記上書き描画処理を実行するようによい。

【0012】

また、第2構成のプログラムにおいて、コンピュータを、対象オブジェクトの各視点画像について、対象オブジェクトの隣側のオブジェクトの位置を、対象オブジェクトのずらし方向の側に、位相ずらし量以上の分だけ、ずらして描画する手段として機能させるようにしてもよい。また、コンピュータを、隣側のオブジェクトの位置のずらし描画処理を行うことを示す記述がファイル内に存在するときに隣側のオブジェクトの位置のずらし描画処理を実行する手段として機能させてもよい。

【0013】

これらのプログラムにおいて、コンピュータを、ファイルに位相ずらし量を示す情報として程度情報が記述されている場合に、予め保持している設定テーブルの情報と前記程度情報とから位相ずらし量を算出する手段として機能させるようになっていてもよい。また、コンピュータを、対象オブジェクトが手前側に立体視されるようにずらし方向が設定されているときには対象オブジェクトを拡大描画処理し、奥側に立体視されるようにずらし方向が設定されているときには、対象オブジェクトを縮小描画処理する手段として機能させるようになっていてもよい。

【0014】

【発明の実施の形態】

以下、この発明の映像ファイル処理方法を図1乃至図13に基づいて説明していく。

【0015】

図1にパーソナルコンピュータのアーキテクチャの一例を示す。CPU1はシステムコントロール機能を持つノースブリッジ2とPCIバスやISAバスなどのインタフェース機能を持つサウスブリッジ3に接続される。ノースブリッジ2には、メモリ4や、AGP (Accelerated Graphics Port) を介してビデオカード5が接続される。そして、サウスブリッジ3には、USB (Universal Serial Bus) インタフェース6、ハードディスクドライブ (HDD) 7、及びCD-ROM装置8等が接続される。

【0016】

図2に一般的なビデオカード5を示す。VRAM（ビデオメモリ）コントローラ5bはAGPを介してCPU1からの命令で描画データのVRAM5aへの書き込み・読み込みの制御を行う。DAC（D/A変換器）5cはVRAMコントローラ5bからのデジタル映像データをアナログ映像信号に変換し、この映像信号をビデオバッファ5dを介してパソコン用モニタ12に供給する。かかる映像表示処理（描画処理）において、右眼映像と左眼映像とを生成し、これらを交互に縦ストライプ状に描画するなどの立体映像表示処理が行える。

【0017】

パーソナルコンピュータはネット接続環境を備え、インターネット上のサーバなどとして構成される送信側装置から例えばHTMLファイルやXMLファイルを受信することができる。また、パーソナルコンピュータは、例えば前記モニタ12に液晶バリアを備えることにより、平面視映像の表示及び立体視映像の表示の両方が行なえるようになっている。立体視映像が、例えば、右眼映像と左眼映像とを交互に縦ストライプ状に配置したものであれば、CPU1の制御により、液晶バリアにおいて、縦ストライプ状の遮光領域が形成される。また、画面上の一部領域（ファイル再生のウィンドウ部分、或いは、HTMLファイルのなかの一部映像部分）において立体視映像を表示することとなるのであれば、前記CPU1によって前記ウィンドウや一部映像部分の表示座標及び大きさに基づいて前記縦ストライプ状の遮光領域の大きさ及び形成位置が制御することが可能である。液晶バリアに限らず、通常のバリア（バリアストライプが所定ピッチで固定的に形成されている）を用いても構わない。また、パーソナルコンピュータはブラウザソフトウェア（ビューア）を搭載しており、ファイルを開いてモニタ12に映像を表示することができる。

【0018】

次に、パーソナルコンピュータ（ブラウザソフトウェア）によるファイルの立体視用描画処理を図3及び図4について説明していく。図3では、「うちの近所にテがあります」といった文の表示例を示しており、同図（a）では平面表示例を示し、同図（b）（c）では「テ」の絵文字部分の立体表示（飛び出して見え

る) 例を示している。同図 (b) に示しているごとく、左眼用映像として「テ」の絵文字部分を右側に所定画素分ずらし (このずらしによって「が」の部分が隠れる (侵食される) ことになる)、右眼用映像として「テ」の絵文字部分を左側に所定画素分ずらす (このずらしによって「に」の部分が隠れる (侵食される) ことになる)。ここで、「テ」の絵文字部分のずらし量はファイルの記述に基づいて算出され、図 3 (b) における左眼表示用データ (うちの近所に テあります) 及び右眼表示用データ (うちの近所テ があります) が生成される。「うちの近所にテがあります」の記述開始位置は、ファイル内に記述されている、例えば、`<start_x>100</start_x>` で示される X 座標及び `<start_y>50</start_y>` で示される Y 座標により特定される。そして、左眼表示用データ (うちの近所に テあります) を構成している画素データ及び右眼表示用データ (うちの近所テ があります) を構成している画素データを、前記座標に対応する VRAM の画素データ記憶位置から交互に (表示画像として水平方向に右眼用画素と左眼用画素とを 1 画素ずつ交互に)、書き込んでいく処理を行う。なお、図 3 (c) では、「テ」と「が」の重なり部分及び「テ」と「に」の重なり部分が半透明表示された例を示している。半透明となるように画像を合成するには、両画像データの $1/2$ 値を加算したものを画像データとする処理を行えばよい。

【0019】

図 4 では、「私は今日東京に行きます」といった文の表示例を示しており、同図 (a) では平面表示例を示し、同図 (b) では「東京」部分の立体表示 (飛び出して見える) 例を示している。この図 4 (b) では、「行きます」の平面視表示部分をもずらすことで、隣接文字の隠れ (侵食) を無くすようにしている。

【0020】

図 5 (a) は「私は今日東京に行きます」といった文の平面視表示のためのファイルの記述例を示しており、同図 (b) は「私は今日東京に行きます」といった文における「東京」部分の立体視表示のためのファイルの記述例 (侵食なし) を示している。パーソナルコンピュータ (ブラウザソフトウェア) は、ファイル内の記述のなかで立体視用処理を示す記述部分を判断する。図 5 の例では `<3d`

>と</3d>とで囲まれた部分が立体視用処理を示す記述部分であると判断することになる。そして、立体視用処理を示す記述部分に基づいて対象オブジェクト（対象文字，対象画像）の位相ずらし量及びずらし方向を判断する。図の例では、「東京」が対象文字であると認識する。そして、<zuras i__L__X>8</zuras i__L__X>とあり、この記述に基づいて左眼用文字の位相ずらし量は右側に8画素であると判断する。また、<zuras i__R__X>,</zuras i__R__X>とあり右眼用文字のずらし量が同様に定義可能であるが、ここでは省略（実際には記載しない）されており、デフォルト値（0）であると判断する。このように判断した前記位相ずらし量及びずらし方向に基づいて対象文字である「東京」の右眼用画像描画及び左眼用画像描画を行うことになる。なお、この例は侵食無しの場合であり、パーソナルコンピュータ（ブラウザソフトウェア）は、対象オブジェクトの隣側のオブジェクト（文字又は画像）の位置を、対象オブジェクトのずらし方向の側に、位相ずらし量以上の分だけ、ずらして描画する処理を行う。図5（b）の例では、「に行きます」の後に記述されている<zuras i__X>,</zuras i__X>により（この記述は実際には省略されている。かかる場合は、<zuras i__L__X>の値、すなわち、8が設定される）、「に行きます」の文字を全体として8画素だけ右側にシフト描画することになる。

【0021】

このように、パーソナルコンピュータ（ブラウザソフトウェア）は、上記記述の意味するところに従って、対象オブジェクト、位相ずらし量、及びずらし方向を判断し、右眼用画像描画及び左眼用画像描画の処理を行うことになる。

【0022】

なお、平面視表示させる部分（「に行きます」）の位相ずらし量は左眼用画像と右眼用画像とで等しくする必要がある。立体視表示部分（「東京」）は、飛び出し量（或いは奥まり量）に応じてずらし量を変更することになる。また、立体視表示部分と平面視表示部分とのずらし量が異なっても良い。ただし、（3D表示部分の位相ずらし量）≤（2D表示部分の位相ずらし量）としないと、侵食が生じてしまうことになる。また、各オブジェクト（「東」「京」）の立体表

示をより効果的に行うために、オブジェクトを分離して（上記では「東」と「京」）、個別に位相ずらし量を与えてもよい。

【0023】

図6は他の記述例（侵食の有無をタグで表す）を示している。＜shift＞8</shift>は、左眼用画像と右眼用画像の相対位相ずらし量が+8であることを示している。＜overlap>有</overlap>は侵食有りを定義している。

【0024】

左眼用画像と右眼用画像の相対位相ずらし量

＋（正：飛出し）： 左眼用文字を右眼用文字に対して右にずらす（8画素ずらす）。

－（負：奥まり）： 右眼用文字を左眼用文字に対して右にずらす（8画素ずらす）。

【0025】

侵食の有無

有： 「東京」を左右均等にずらす（4画素ずつずらす）。

「に行きます」は、ずらさない。

無： 「東京」を右にだけずらす（8画素ずらす）。

「に行きます」を左右文字の相対位相ずらし量と同じ値（絶対値）右にずらす（8画素ずらす）。

すなわち、パーソナルコンピュータ（ブラウザソフトウェア）は、上記記述の意味するところに従って、対象文字、位相ずらし量、及びずらし方向を判断し、右眼用画像描画及び左眼用画像描画の処理を行う。

【0026】

図7は他の記述例（飛び出し量をレベルで表す）を示している。

【0027】

立体方向

飛出し： 左眼用文字を右眼用文字に対して右にずらす。

奥まり： 右眼用文字を左眼用文字に対して右にずらす。

【0028】

左右文字の相対位相ずらしレベル

レベルと画素数との関係が定義される。定義に対応する具体的画素数は、予めパーソナルコンピュータのメモリに設定テーブルとして格納されている。

強： 8画素ずらす

中： 4画素ずらす

弱： 2画素ずらす

【0029】

侵食の有無

有： 「東京」を左右均等にずらす（4画素ずつずらす）。

「に行きます」は、ずらさない。

無： 「東京」を右にだけずらす。（8画素ずらす）

「に行きます」を左右文字の相対位相ずらし量と同じ値（絶対値）右にずらす（8画素ずらす）。

すなわち、パーソナルコンピュータ（ブラウザソフトウェア）は、上記記述の意味するところに従って、対象文字、位相ずらし量、及びずらし方向を判断し、右眼用画像描画及び左眼用画像描画の処理を行う。

【0030】

図8は他の記述例（飛び出し量を割合で表す）を示している。 $\langle \text{shift_ratio} \rangle 60 \langle /\text{shift_ratio} \rangle$ は、左眼用画像と右眼用画像の相対位相ずらし率が60%であることを示している。

【0031】

立体方向

飛び出し： 左眼用文字を右眼用文字に対して右にずらす。

奥行き： 右眼用文字を左眼用文字に対して右にずらす。

【0032】

左右文字の相対位相ずらし最大値は10とされる。この情報は予めパーソナルコンピュータのメモリに設定テーブルとして格納されている。

【0033】

N%: $10 \times N / 100$ 画素ずらす (N=60 なら 6 画素ずらす)。

小数点以下は四捨五入し、整数とする。

最大値はユーザが指定してもよいし、ディスプレイ情報 (ディスプレイのサイズによって最大値が異なる) であってもよい。

【0034】

侵食の有無

有: 「東京」を左右均等にずらす (3 画素ずつずらす)。

「に行きます」は、ずらさない。

無: 「東京」を右にだけずらす (6 画素ずらす)。

「に行きます」を左右文字の相対位相ずらし量と同じ値 (絶対値) 右にずらす (6 画素ずらす)。

すなわち、パーソナルコンピュータ (ブラウザソフトウェア) は、上記記述の意味するところに従って、対象文字、位相ずらし量、及びずらし方向を判断し、右眼用画像描画及び左眼用画像描画の処理を行う。

【0035】

図 9 は他の記述例 (動きを表す) を示している。

【0036】

動き

前後: 標準位相ずらし量でスタートし、最大位相ずらし量、最小位相ずらし量の間で前後往復移動。

左右: 標準位相ずらし量で、move__h の半分の位置からスタートし、左右往復移動。

左から右: 左から右に移動し、右に到達すると、再度左から移動を開始 (移動量は move__h)

右から左: 右から左に移動し、左に到達すると、再度右から移動を開始 (移動量は move__h)

回転: 正面 (2D) 状態でスタートし、後述の捕捉説明の要領で回転

静止: 標準位相ずらし量で、静止 (動きなし)

他にも、前から後、後から前も可能。

【0037】

動きの回数

指定回動き表示し、スタート状態に戻って停止する。0を指定した場合は、無限に繰り返す。

【0038】

侵食の有無

有: 「東京」を左右均等にずらす。

「に行きます」は、ずらさない。

無: 「東京」を右にだけずらす。

「に行きます」を、`typ__shift`、`max__shift`、`min__shift`の絶対値のうち、最も大きい値分、右にずらす。左右移動の場合は、さらに`move__h`を加算する。

【0039】

なお、動きの種類によって使用するタグが変わるので、不要なタグが記述されている場合、パーソナルコンピュータ（ブラウザ）は無視する。

【0040】

補足1: 回転の方法

回転の最も簡単な実現方法は、通常 of 文字 of 回転を利用して異なる角度で文字を取り出す手法である。図10に示すように回転する文字は、「東京」という文字を異なる角度から見た図形といえるので、微妙に異なる角度 of 二つの文字図形を左右画像とすることにより、立体視が可能となる。このとき、「東京」という文字の中に、飛び出しの部分と奥まり of 部分が存在するので、文字 of 部位（画素）によって位相ずらし量が変わっていることになる。これらの状態のうち最も視差が強くなる状態を、`max__shift`、`min__shift`で指定する。この値から、文字 of 部位 of 描画位置を決めることができる。

【0041】

補足2: 立体方式

以上の事柄は立体方式にかかわらず実現可能である。視点数が2より大きい場合、例えば4視点 of 場合は、第1視点文字と第2視点文字と第3視点文字と第4

視点文字の位相ずらし量を決める。また、多視点では侵食しない場合の処理が変わる。4視点の場合は、第1視点文字と第4視点文字のずらし分（左右文字の相対位相ずらし量×3）、「に行きます」をずらす。ただし、このように立体の方式や視点数によって合成処理や侵食時の処理が変わるので、そのような情報を予め与えておけばよい。合成処理のための内容は関数で与えることもできる。

すなわち、パーソナルコンピュータ（ブラウザソフトウェア）は、上記記述の意味するところや視点数情報や関数等に従って、対象文字、位相ずらし量、及びずらし方向等を判断し、第1視点文字と第2視点文字と第3視点文字と第4視点文字の描画の処理を行う。

【0042】

図11（a）は他の記述例（侵食の有無及び透明度をタグで表す）を示しており、同図（b）は描画処理を示した説明図である。

【0043】

左右文字の相対位相ずらし量

＋（正：飛出し）： 左眼用文字を右眼用文字に対して右にずらす（8画素ずらす）。

－（負：奥行き）： 右眼用文字を左眼用文字に対して右にずらす（8画素ずらす）。

【0044】

侵食の有無（<overlap>）

有： 絵文字を左右均等にずらす（4画素ずつずらす）。

「に行きます」は、ずらさない。

無： 絵文字を右にだけずらす（8画素ずらす）。

「に行きます」を左右文字の相対位相ずらし量と同じ値（絶対値）右にずらす（8画素ずらす）。

【0045】

表示優先度（<priority>）

1：立体表示する文字を上レイヤーにして表示

（侵食される平面表示される文字が隠れる）

2：立体表示する文字を下のレイヤーにして表示

0：表示するブラウザ（プレイヤ）に依存

このタグを設定しない場合（デフォルト）は0

【0046】

透明度 ($\langle \text{transparency} \rangle M \langle / \text{transparency} \rangle$) の記述により、表示する文字（絵文字）の透明度がM%で設定される。デフォルト（未設定時）は0%とする。例えば、透明指定されたオブジェクトのR画素のデータをR1、これと重なるオブジェクトのR画素のデータをR2とすると、描画R画素データは、 $(R2 \times M / 100 + R1 \times (1 - M / 100))$ のように演算することで得られる。

【0047】

ところで、オブジェクトを飛び出させて立体視させる場合、図12に示すように、体感文字幅Fは本来の文字幅Dよりも小さくなる。そこで、ファイル記述における文字（指定サイズ文字）の平面視表示では図13（a）となるような文字列に対して、立体視の対象となる「東京」の文字をブラウザが例えば2倍の文字サイズに加工する処理を実行する。この処理の後、右眼用画像と左眼用画像の描画処理（位相ずらし処理）を行うことになる。侵食無しの設定であれば、図13（b）に示すように、「に行きます」の位相もずらすことになる。一方、侵食有りの設定であれば、図13（c）に示すように、描画処理を行う。

【0048】

図12から分かるように、 $D : (A + B) = F : B$ 、 $C : A = E : B$ の関係があり、 $F / D = E / (E + C)$ となる。飛出しにより文字が $E / (E + C)$ 倍になる（縮む）ため、予め文字を $(E + C) / E$ 倍に拡大して表示する。Eは65mm程度の定数である。例えば視差量 $C = 65 \text{ mm}$ に相当する位相ずらし量が設定される場合は、文字が $1 / 2$ に縮小されて体感されるため、予め文字を2倍にして描画することになる。ここで、パーソナルコンピュータ（ブラウザ）は、自身のモニタ12の画素ピッチ情報（画面インチサイズ及び画面解像度によって画素ピッチが得られるテーブルを持ち、例えば、ユーザによって画面インチサイズ及び画面解像度を入力させることで、画素ピッチ（mm）が得られる）を保持し

ておく。パーソナルコンピュータ（ブラウザ）は、ファイル記述における位相ずらし量（画素数）に前記画素ピッチを乗算して求めた C （mm）と $E = 6.5 \text{ mm}$ とにより、 $(E + C) / E$ の値を求め、この値に基づいて元来の文字に対する画素補間（拡大処理）を行う。或いは $(E + C) / E$ の値を元来の文字サイズに乗算して得られる大きさを満たす文字サイズを判断し、この文字サイズの「東京」のドットデータを取得して描画する。この描画の際、拡大した「東京」の上下位置が行の中央になるように座標を決めて描画する。例えば、元来の文字の縦画素数が 20 で拡大文字の縦画素数が 40 となる時、 $(40 - 20) / 2 = 10$ の演算により、基準位置（下詰め位置）に対して縦方向に「東京」の文字を 10 画素分だけ、下にずらして描画する。

【0049】

なお、上記の例では、ファイルを認識して画像表示するパーソナルコンピュータを例示したが、これに限るものではなく、データ放送（BML ファイル）を受信して画像表示できるデジタル放送受信装置や、ネット接続環境及び画像表示機能を備える携帯電話などとしても構成できる。

【0050】

【発明の効果】

以上説明したように、この発明によれば、HTML ファイルなどのファイルに基づいて任意の文字部分や画像部分を立体表示させることができるという効果を奏する。

【図面の簡単な説明】

【図 1】

パーソナルコンピュータのアーキテクチャ例を示したブロック図である。

【図 2】

ビデオカードの構成例を示したブロック図である。

【図 3】

パーソナルコンピュータ（ブラウザソフトウェア）によるファイルの立体視用描画処理の説明図である。

【図 4】

パーソナルコンピュータ（ブラウザソフトウェア）によるファイルの立体視用
描画処理の説明図である。

【図 5】

ファイルの記述例を示した説明図である。

【図 6】

ファイルの記述例を示した説明図である。

【図 7】

ファイルの記述例を示した説明図である。

【図 8】

ファイルの記述例を示した説明図である。

【図 9】

ファイルの記述例を示した説明図である。

【図 10】

回転画像の例を示した説明図である。

【図 11】

ファイルの記述例を示した説明図である。

【図 12】

立体視の原理を示した説明図である。

【図 13】

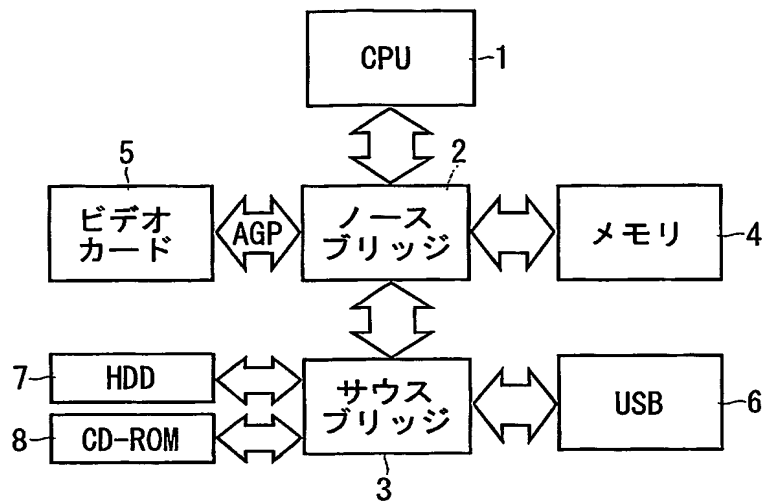
同図（a）は平面視表示例であり、同図（b）（c）は立体視する文字のサイズを拡大する立体視表示例を示した説明図である。

【符号の説明】

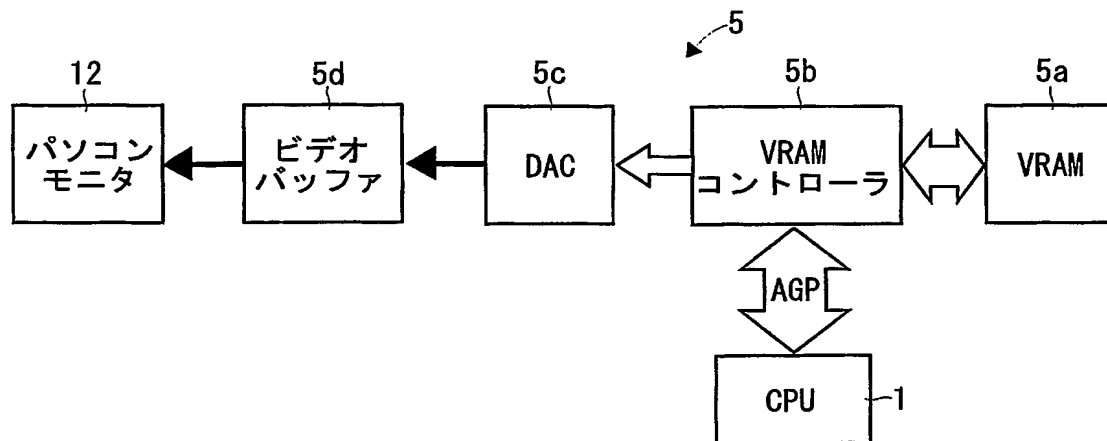
- 1 CPU
- 4 メモリ
- 5 ビデオカード
- 5 a VRAM
- 5 b VRAMコントローラ

【書類名】 図面

【図 1】



【図 2】






【図 3】




(a)

2D表示 うちの近所に  があります

(b)

3D表示 (左眼) うちの近所に  あります
(右眼) うちの近所  があります
(立体) うちの近所に  があります

(c)

3D表示
(透明度設定あり) (左眼) うちの近所に  あります
(右眼) うちの近所  があります
(立体) うちの近所に  があります

【図 4】

(a)

2D表示 私は今日東京に行きます

(b)

3D表示 (左眼) 私は今日 東京に行きます
(右眼) 私は今日東京 に行きます
(立体) 私は今日 東京 に行きます

【図 5】

(a)

2 D

<word>	文字列表示を明記するタグ
<start_x> 100 </start_x>	表示開始位置の X 座標
<start_y> 50 </start_y>	表示開始位置の Y 座標
<nationality> 日本語 </nationality>	表示言語
 MSゴシック 	使用フォント
<size> 12 </size>	使用文字サイズ
<hyouji> 私は今日東京に行きます </hyouji>	表示文書
</word>	

(b)

3 D

<word>	文字列表示を明記するタグ
<start_x> 100 </start_x>	表示開始位置の X 座標
<start_y> 50 </start_y>	表示開始位置の Y 座標
<nationality> 日本語 </nationality>	表示言語
 MSゴシック 	使用フォント
<size> 12 </size>	使用文字サイズ
<hyouji>	表示文書
私は今日	
<3d>	3 D 表示する文字列
東京	左眼用文字の位相ずらし量
<zurashi_L_x> 8 </zurashi_L_x>	(デフォルト値: 0、右側が+)
(<zurashi_R_x> , </zurashi_R_x>)	(右眼用文字の位相ずらし量)
	(デフォルト値: 0、左側が+)
</3d>	
に行きます	位相ずらし量
(<zurashi_x>, </zurashi_x>)	(設定しない場合は、
	上記<zurashi_L_x>の設定値)
</hyouji>	
</word>	

【図 6】

3D

<code><word></code>	文字列表示を明記するタグ
<code><start_x> 100 </start_x></code>	表示開始位置の X 座標
<code><start_y> 50 </start_y></code>	表示開始位置の Y 座標
<code><nationality> 日本語 </nationality></code>	表示言語
<code> MSゴシック </code>	使用フォント
<code><size> 12 </size></code>	使用文字サイズ
<code><sentence></code>	表示文書
私は今日	
<code><3d></code>	3D表示する文字列
東京	左右文字の相対位相ずらし量
<code><shift> 8 </shift></code>	(デフォルト値: 0、飛出しが+)
<code><overlap> 有 </overlap></code>	侵食の有無
<code></3d></code>	
に行きます	
<code></sentence></code>	
<code></word></code>	

【図 7】

3D

<word>	文字列表示を明記するタグ
<start_x> 100 </start_x>	表示開始位置の X 座標
<start_y> 50 </start_y>	表示開始位置の Y 座標
<nationality> 日本語 </nationality>	表示言語
 MSゴシック 	使用フォント
<size> 12 </size>	使用文字サイズ
<sentence>	表示文書
私は今日	
<3d>	3D表示する文字列
東京	立体方向
<direction> 飛出し </direction>	左右文字の相対位相ずらしレベル
<shift_level> 強 </shift_level>	侵食の有無
<overlap> 有 </overlap>	
</3d>	
に行きます	
</sentence>	
</word>	

【図 8】

3D

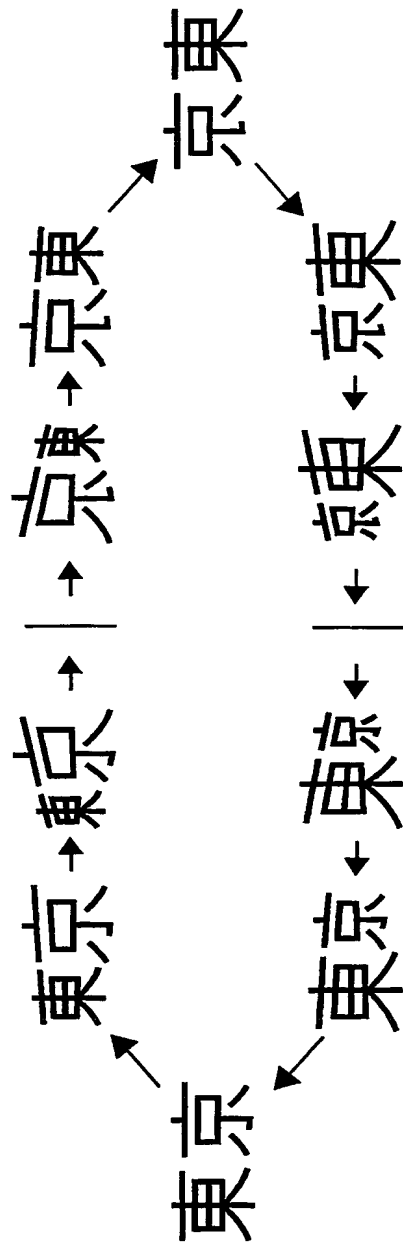
<code><word></code>	文字列表示を明記するタグ
<code><start_x> 100 </start_x></code>	表示開始位置の X 座標
<code><start_y> 50 </start_y></code>	表示開始位置の Y 座標
<code><nationality> 日本語 </nationality></code>	表示言語
<code> MSゴシック </code>	使用フォント
<code><size> 12 </size></code>	使用文字サイズ
<code><sentence></code>	表示文書
私は今日	
<code><3d></code>	3D表示する文字列
東京	立体方向
<code><direction> 飛出し </direction></code>	立体方向
<code><shift_ratio> 60 </shift_ratio></code>	左右文字の相対位相ずらし率(%)
<code><overlap> 有 </overlap></code>	侵食の有無
<code></3d></code>	
に行きます	
<code></sentence></code>	
<code></word></code>	

【図 9】

3D

<word>	文字列表示を明記するタグ
<start_x> 100 </start_x>	表示開始位置の X 座標
<start_y> 50 </start_y>	表示開始位置の Y 座標
<nationality> 日本語 </nationality>	表示言語
 MSゴシック 	使用フォント
<size> 12 </size>	使用文字サイズ
<sentence>	表示文書
私は今日	
<3d>	3D表示する文字列
東京	動きの種類
<motion> 前後 </motion>	動きの種類
<typ_shift> 1 </typ_shift>	左右文字の標準位相ずらし量
<max_shift> 8 </max_shift>	左右文字の最大位相ずらし量
<min_shift> -8 </min_shift>	左右文字の最少位相ずらし量
	(いずれもデフォルト値: 0、飛出し+)
<move_h> 5 </move_h>	左右移動量
<frequency> 1 </frequency>	動きの周波数
	(回/秒、往復/秒、回転/秒)
<repeat> 10 </repeat>	動きの回数
<overlap> 有 </overlap>	侵食の有無
</3d>	
に行きます	
</sentence>	
</word>	


【図 10】




【図 11】

(a)

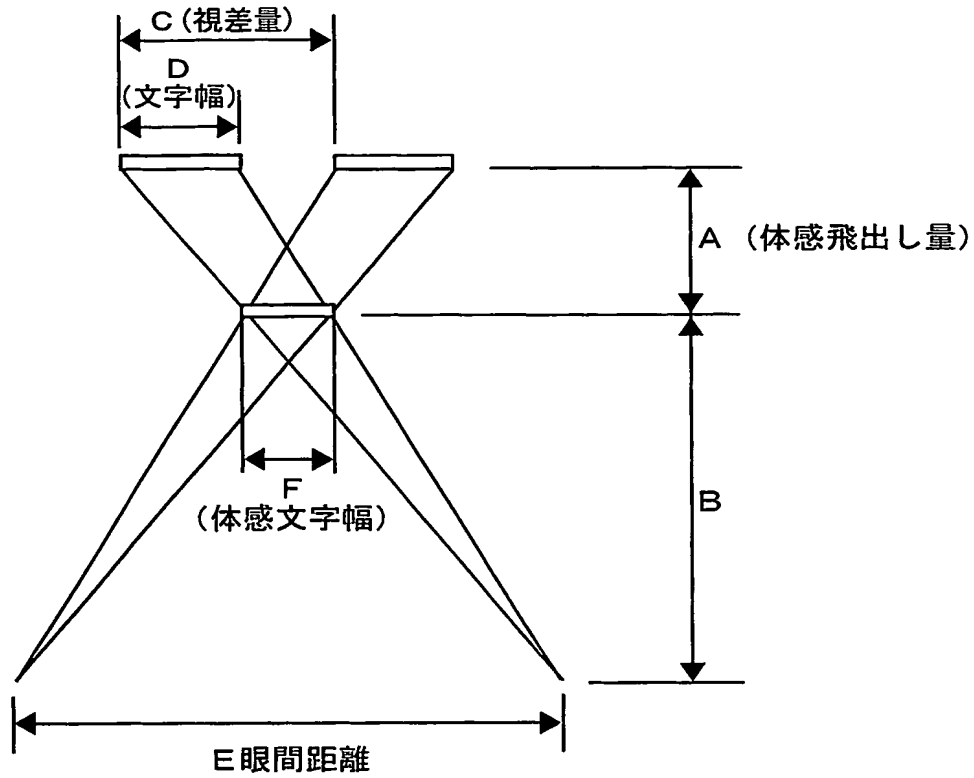
3D

<word>	文字列表示を明記するタグ
<start_x> 100 </start_x>	表示開始位置の X 座標
<start_y> 50 </start_y>	表示開始位置の Y 座標
<nationality> 日本語 </nationality>	表示言語
 MSゴシック 	使用フォント
<size> 12 </size>	使用文字サイズ
<sentence>	表示文書
私は今日	
<3d>	3D表示する文字列
	左右文字の相対位相ずらし量 (デフォルト値: 0、飛出し+)
<shift> 8 </shift>	
<overlap> 有 </overlap>	侵食の有無
<priority> 1 </priority>	表示優先度の有無
<transparency> 50 </transparency>	透明度 (%)
</3d>	
に行きます	
</sentence>	
</word>	

(b)

(左眼) 私は今日  に行きます(右眼) 私は今日  に行きます(3D) 私は今日  に行きます

【図 12】



【図 13】

(a)

2D表示 私は今日東京に行きます

サイズ: 20

(b)

文字拡大 私は今日**東京**に行きます

「東京」サイズ: 40

(c)

文字拡大 私は今日**東京**に行きます

「東京」サイズ: 40

【書類名】 要約書

【要約】

【目的】 HTMLファイルなどのファイルに基づいて任意の文字部分や画像部分を立体表示させることができる立体映像表示装置を提供する。

【構成】 「〒」の絵文字部分のずらし量はファイルの記述に基づいて算出され、図3（b）における左眼表示用データ（うちの近所に 〒あります）及び右眼表示用データ（うちの近所〒 があります）が生成される。「うちの近所に〒があります」の記述開始位置は、ファイル内に記述されているX座標及びY座標により特定される。そして、左眼表示用データ（うちの近所に 〒あります）を構成している画素データ及び右眼表示用データ（うちの近所〒 があります）を構成している画素データを、前記座標に対応するVRAMの画素データ記憶位置から交互に（表示画像として水平方向に右眼用画素と左眼用画素とを1画素ずつ交互に）、書き込んでいく処理を行う。

【選択図】 図3

特願 2 0 0 3 - 1 5 0 0 8 8

出 願 人 履 歴 情 報

識別番号 [0 0 0 0 0 1 8 8 9]

1. 変更年月日 1 9 9 3 年 1 0 月 2 0 日

[変更理由] 住所変更

住 所 大阪府守口市京阪本通 2 丁目 5 番 5 号

氏 名 三洋電機株式会社